# ディープラーニングフレームワーク Caffe の分散環境への適用

一瀬絢衣　(指導教員：小口正人)

## 1　はじめに

近年インターネット上の情報量の増大やクラウドコンピューティングの普及により，ライフログの取得やそのデータの蓄積が容易になった．その結果監視カメラなどのセンサを用いたライフログの利用も普及してきている．ここで，一般家庭にサーバやストレージを設置して解析までの処理を行うことは困難であるが，映像をそのまま送信しクラウドで処理する場合，プライバシや各種センサとクラウド間のネットワーク帯域の問題が生じてしまう．

本研究では，データ量の削減とプライバシの保護を目的とし，ディープラーニングのフレームワークである Caffe を分散環境へ適用させることを考える．

ここで，処理を分散する際のデータ量の削減は識別率を下げてしまうと考えられるため，パラメータを変化させてデータ量を削減させた際の識別率の調査と，処理時間の比較による評価を行った．

## 2　ディープラーニング

ディープラーニングとは，人間の脳の神経回路がもつ仕組みを模した情報処理システムであるニューラルネットワークの中で，識別を行う中間層を多層化したものを用いた機械学習を指す．中間層が複数になっていることにより何段階かで認識を繰り返し，色や形状，全体像など複数の特徴を抽出してより正確な識別が可能となっている．

そのフレームワークである Caffe は，Convolutional Architecture for Fast Feature Embedding の略であり，Berkeley Vision and Learning Center が中心となって進めている．C++で実装されており，GPU に対応していることから高速な処理が可能であること，学習済みネットワークモデルが提供されていて簡単に実験を行うことができるという特徴がある．

Caffe は，ディープラーニングの中でも畳込みニューラルネットと呼ばれる構造になっている．畳込みニューラルネットは主に画像認識に応用されるネットワークである．通常畳込みとプーリングという画像処理の基本的な演算を行う層がペアで複数回繰り返されたあと，全結合層が配置される構造になっている．畳込み層では入力画像に対しフィルタを適用し，フィルタをずらしながら各重なりで両者の積和計算を行うことによってフィルタが表す特徴的な濃淡構造を画像から抽出する．プーリング層では画像上で正方領域をとり，この中に含まれる画素値を使って一つの画素値を求め，畳込み層で抽出された特徴の位置感度を若干低下させる．

## 3　分散ディープラーニングフレームワーク

本研究では，図 1 で示すフレームワークを提案する．畳込みニューラルネットワークに新たな層を定義してネットワークを分離し，クライアント側，クラウド側を用いた分散処理を実現する．分散処理を行うことにより，映像や画像そのものでなく特徴量をクラウドに送るためプライバシが確保され，またデータ量を小さくしてから送ることによりネットワーク帯域を考慮した処理が可能になると考えられる．

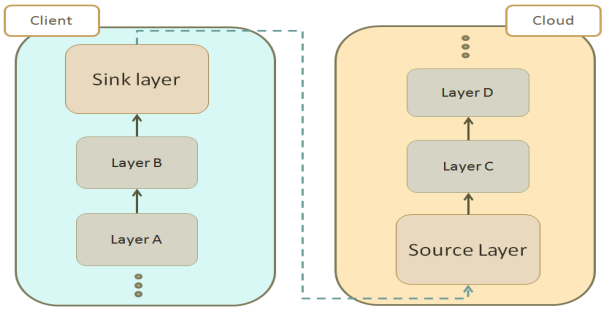


図 1: 分散ディープラーニングフレームワーク

## 4　基礎実験

### 4.1　データセット

実験では，CIFAR-10 データセットを用いた．

CIFAR-10 とは 32 × 32 画素の画像が 10 種類のクラスに分類されているデータセットであり，Caffe で学習済みネットワークモデルが提供されているデータセットの一つである．このネットワークモデルにおいて各層で定義されているパラメータは表 1 のようになっている．

Caffe においてデータはバッチサイズ，チャネル数，画像の大きさの 4 次元配列でデータが格納されており，チャネル数は直前の畳込み層におけるフィルタ数と一致する．提供されているネットワークモデルのデフォルトの値では，始めは (100 × 3 × 32 × 32)byte のデータ量から conv1 層を通り (100 × 32 × 32 × 32)byte へ，ストライドを 2 に定義している pool1 層を通り (100 × 32 × 16 × 16)byte へと変化しており，pool3 層までは最初のデータ量よりも大きくなっていることがわかる (図 2)．

表 1: パラメータ

| | |
|---|---|
| n：num_output | フィルタの数 |
| p：pad | パディングの幅 |
| k：kernel_size | フィルタの大きさ |
| s：stride | フィルタの適用位置の間隔 |

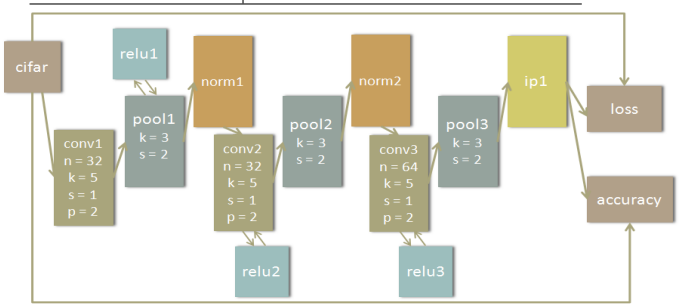


図 2: CIFAR-10 識別用ネットワークモデル

### 4.2 実験概要

畳込み層におけるフィルタ数を変化させることによってクライアントからクラウドへ送る際のデータ量を小さくすることを検討する．その際，データ量を小さくすると識別率が低くなってしまうことが考えられるため，フィルタ数を変化させて識別率がどう変化するのかを調査するため，conv2 層のフィルタ数を 1 からデフォルト値の 32 まで変化させ，通信時のデータ量と識別率の相関を示した．実験環境は表 2 に示す．

表 2: 実験環境

| | |
|---|---|
| OS | Ubuntu 14.04LTS |
| CPU | Intel(R) Xeon(R) CPU W5590 @3.33GHz (8 コア) × 2 ソケット |
| Memory | 8Gbyte |

### 4.3 実験結果

結果を図 3 に示す．少ないデータ量で識別率が収束していることが確認でき，フィルタ数を削減しても高い識別率を保てるということがわかった．フィルタ数が 8 の場合に 0.7656 を計測しているが，これは画像をそのまま送信する場合と比較して 6 分の 1 のデータ量でクライアントからクラウドへ送信することができる．

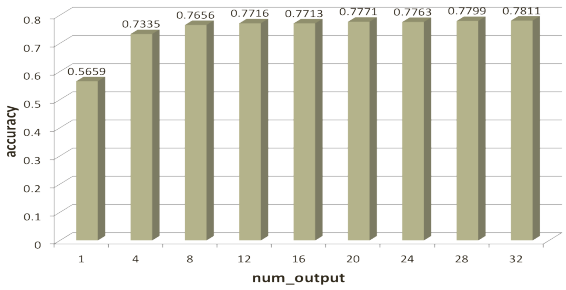


図 3: conv2 層のフィルタ数を変化させた場合の識別率

## 5 評価実験

### 5.1 実験概要

分散処理の評価を行うにあたり，conv2 層のフィルタ数を変化させたときの以下の 3 つの場合の処理時間を計測し，比較を行った．処理時間は，CIFAR-10 の評価を行った際の 1 バッチサイズにおける処理時間を表している．実験環境は基礎実験と同じであり，クライアント，クラウド間のネットワーク帯域は 1Gbps である．

実験 1 すべての処理をクライアント側で行うことを想定して CPU のみでの処理時間を計測した．

実験 2 norm1 層後にデータをクラウドへ送信することを想定し，conv2 層までの処理を CPU のみで行い，通信時間を含めたクライアント側の処理時間を計測した．

実験 3 すべての処理をクラウドへ送信することを想定し，通信時間を含めた GPU での処理時間を計測した．

### 5.2 実験結果

結果を図 4，通信時間を 100 倍にしたものを図??に示す．

提案する分散処理はすべての処理をクライアントで行う場合と比較すると，クライアント側での処理が減るため 1 バッチサイズの処理時間は削減することができた．ネットワーク帯域が広く通信時間が非常に短いためすべての処理をクラウドへ送信してから送る場合と比較すると処理時間は長くなっていが，想定する環境ではネットワーク帯域は狭い可能性が高く，通信に時間がかかることが予測される．また，プライバシを考慮すると，提案手法は有効であると考えられる．

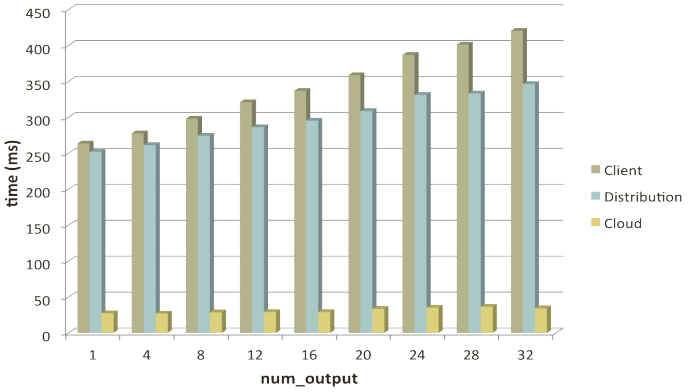


図 4: 処理時間の比較

## 6 まとめと今後の課題

本研究ではディープラーニングのフレームワークである Caffe を分散環境へ適用させることにより，プライバシやネットワーク帯域を考慮したセンサデータ解析処理を検討した．畳込み層におけるフィルタ数を変化させることにより識別率を大幅に下げることなくクライアントからクラウドへ送る際のデータサイズの小さくできることが確認でき，それによって通信時間を削減することができるため，ネットワーク帯域の狭い環境では提案手法が有効であると予測することができた．

今後の課題としては，分散環境におけるネットワーク帯域を考慮した処理時間の比較を行うことを検討している．

## 謝辞

本研究を進めるにあたり，国立研究開発法人産業技術総合研究所の竹房あつ子氏，中田秀基氏に御指導，御助言賜りました．深く感謝いたします．

## 参考文献

[1] 一瀬絢衣，竹房あつ子，中田秀基，小口正人：「ディープラーニングフレームワーク Caffe の分散環境への適用」，
DEIM2016，D1-3，2016 年 3 月発表予定